Shallow Methods for RTE

in

HS Linguistic Inference and Textual Entailment

Prof. Dr. Pinkal

Sommersemester 2007

Michael Wirth
Universität des Saarlandes
Fakultät 4.7 Allgemeine Linguistik
miwirth@coli.uni-sb.de

Outline

- RTE
- Recognizing Textual Entailment Using Lexical Similarity
- Recognizing Textual Entailment with Tree Distance Algorithms
- Textual Entailment Based on Dependency Analysis and WordNet
- Comparsion / Comments

The Pascal Recognizing Textual Entailment Challenge (RTE)

- GOAL:
  - clearer framework

    abstract generic task -> textual entailment
  - corresponding research communities

Textual Entailment

Textual entailment is a directional relationship between pairs of text expressions.

T (Text) entails H (Hypothesis) if the meaning of *H* can be inferred from the meaning of *T*, as would typically be interpreted by people.

Application Settings

- Information Retrieval (IR)
- Comparable Documents (CD)
- Reading Comprehension (RC)
- Question Answering (QA)
- Information Extraction (IE)
- Machine Translation (MT)
- Paraphrase Acquisition (PP)

Recognizing Textual Entailment Challenge (RTE)

Corpus & Evaluation

- Corpus (RTE I)
  - hand-annotated golden standard
    either *H* is entailed by *T* or not (true / false distinction)
  - development set (567 examples)
  - test set (800 examples)

- Evaluation measures
  - Accuracy

$$\text{accuracy} = \frac{\#correct\text{-}responses\text{-}system}{\#correct\text{-}responses\text{-}golden\text{-}standard}$$

  - Confidence-Weighted Scores (*CWS*)

$$cws = \frac{1}{n}\sum_{i=1}^{n}\frac{\#correct-up-to-rank-i}{i}$$

Recognizing Textual Entailment Challenge (RTE)

Results RTE I

baseline:

accuracy = 0.5
cws = 0.5
f-score = 0.67

First Author (Group)	accu-racy	cws	partial coverage	Word overlap	Statistical lexical relations	WordNet	Syntactic matching	world knowledge	Logical inference
				System description					
Akhmatova (Macquarie)	0.519	0.507		X					X
Andreevskaia (Concordia)	0.519	0.515				X	X		
	0.516	0.52							
Bayer (MITRE)	0.586	0.617			X				
	0.516	0.503	73%					X	X
Bos (Edinburgh & Leeds)	0.563	0.593		X		X		X	X
	0.555	0.586		X					
Delmonte (Venice & irst)	0.606	0.664	62%			X	X		X
Fowler (LCC)	0.551	0.56				X		X	X
Glickman (Bar Ilan)	0.586	0.572			X				
	0.53	0.535							
Herrera (UNED)	0.566	0.575		X	X		X		
	0.558	0.571		X					
Jijkoun (Amsterdam)	0.552	0.559		X	X				
	0.536	0.553		X		X			
Kouylekov (irst)	0.559	0.607		X	X		X		
	0.559	0.585							
Newman (Dublin)	0.563	0.592		X	X				
	0.565	0.6							
Perez (Madrid)	0.495	0.517		X					
	0.7	0.782	19%						
Punyakanok (UIUC)	0.561	0.569					X		
Raina (Stanford)	0.563	0.621			X	X	X		X
	0.552	0.686							
Wu (HKUST)	0.512	0.55			X		X		
	0.505	0.536							
Zanzotto (Rome-Milan)	0.524	0.557				X	X		
	0.518	0.559							

Recognizing Textual Entailment Using Lexical Similarity (Jijkoun & de Rijke, 2005)

Method:

- “directed“ sentence similarity
  - frequency-based term weighting
  - two different lexical similarity measures
    - dependency-based word similarity (Lin 1998)
    - lexical chains in WordNet (Hirst and St-Onge 1998)

Algorithm

let $T = (T_1, T_2, \ldots, T_n)$
let $H = (H_1, H_2, \ldots, H_m)$
let $\mathit{totalSim} = 0$
let $\mathit{totalWeight} = 0$
for $j = 1 \ldots m$ do
  let $\mathit{maxSim} = \max_i \mathrm{wordsim}(T_i, H_j)$
  if $\mathit{maxSim} = 0$ **then** $\mathit{maxSim} = -1$
  $\mathit{totalSim} \mathrel{+}= \mathit{maxSim} * \mathrm{weight}(H_j)$
  $\mathit{totalWeight} \mathrel{+}= \mathrm{weight}(H_j)$
end for
let $\mathit{sim} = \mathit{totalSim} / \mathit{totalWeight}$
if $\mathit{sim} \geq \mathit{threshold}$ **then return** TRUE
return FALSE

Measures (1/3)

- confidence

$$confidence = \frac{sim - threshold}{1 - threshold}$$

sim ≥ threshold

Measures (2/3)

- *Normalized Inverse Collection Frequency*
  or
  weighting of words

$$\mathrm{weight}(w) = 1 - \frac{\mathrm{ICF}(w) - \mathrm{ICF}_{\mathrm{min}}}{\mathrm{ICF}_{\mathrm{max}} - \mathrm{ICF}_{\mathrm{min}}}$$

$$\mathrm{ICF}(w) = \frac{\#\ \text{occurences of}\ w}{\#\ \text{occurences of all words}}$$

Recognizing Textual Entailment Using Lexical Similarity

Measures (3/3)

- word similarity measures
  - dependency-based word similarity (Lin 1998a)

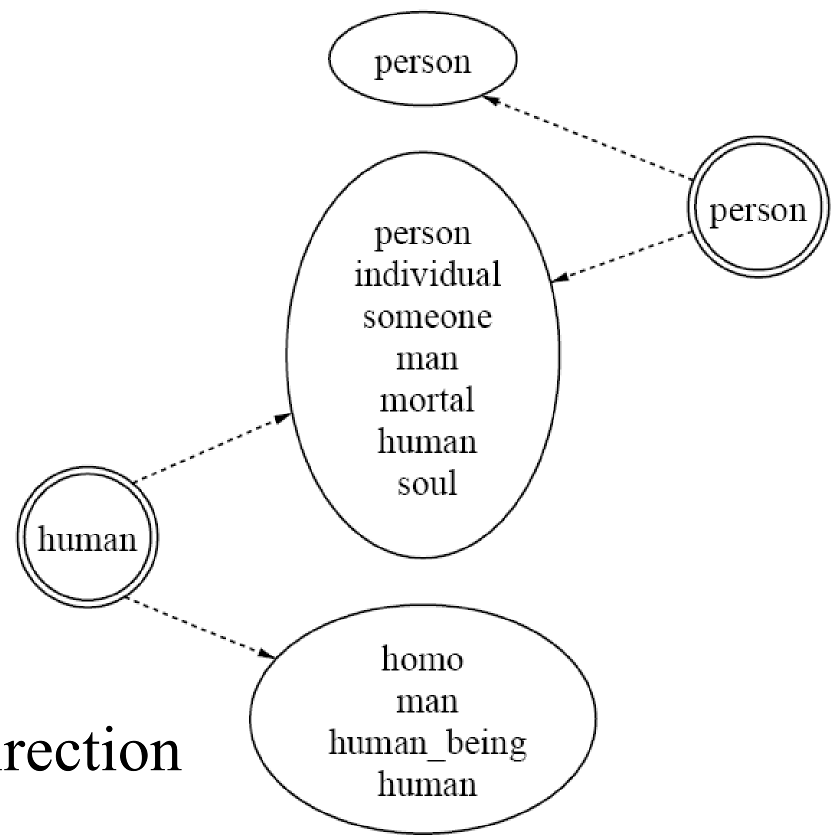
$$\mathrm{sim}(w_1, w_2) = \frac{2 \times I(F(w_1) \cap F(w_2))}{I(F(w_1)) + I(F(w_2))}$$

I(*S*): the amount of information contained in a set of features S
F(*w*): the set of features possesed by *w* extracted from dependency triples

Recognizing Textual Entailment Using Lexical Similarity

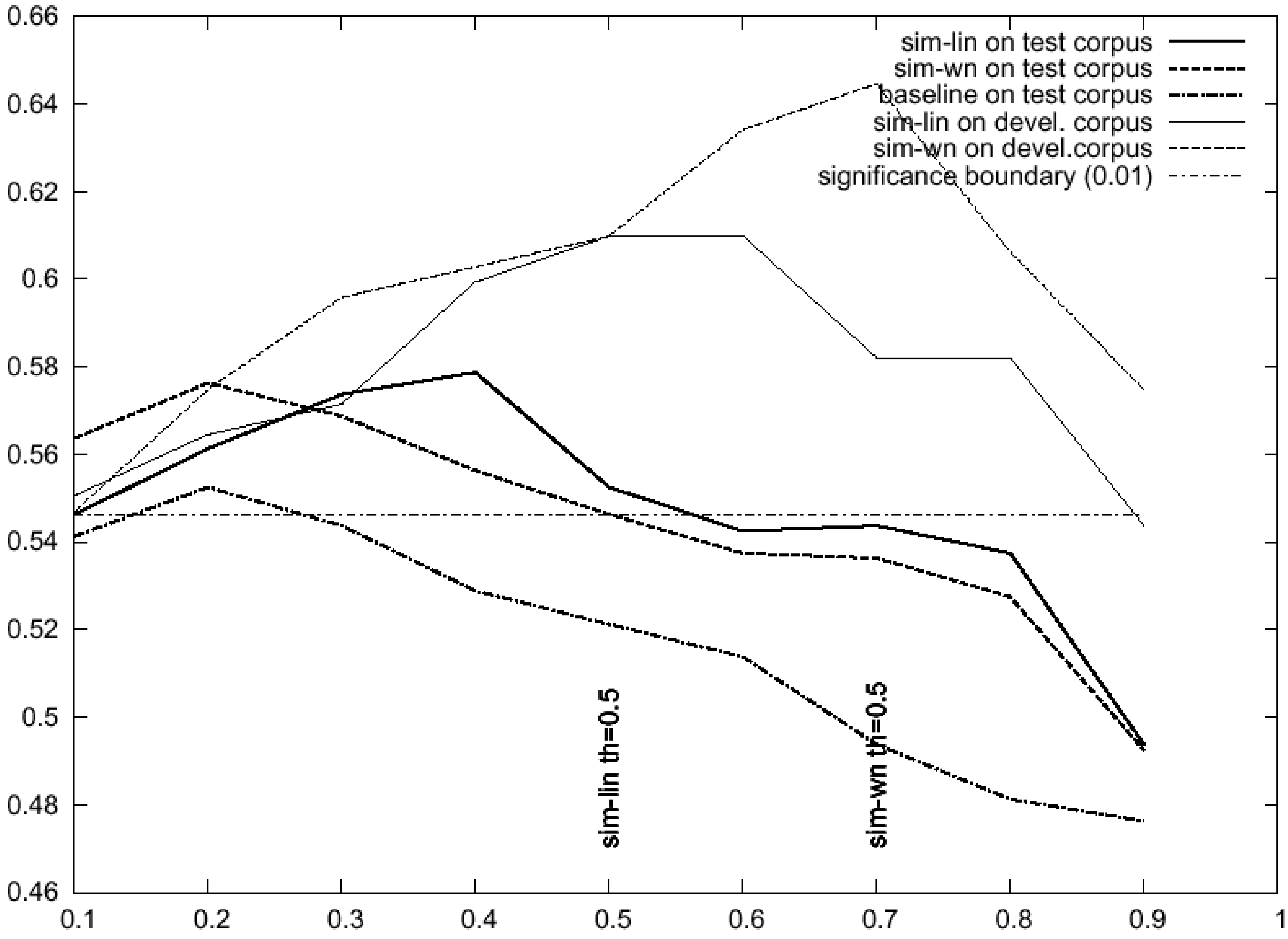
Measures (3/3)

- word similarity measures
  - lexical chains in WordNet



weight = C - path length
- k * number of changes direction

Results


sim-lin on test corpus
sim-wn on test corpus
baseline on test corpus
sim-lin on devel. corpus
sim-wn on devel.corpus
significance boundary (0.01)
sim-lin th=0.5
sim-wn th=0.5
0.66
0.64
0.62
0.6
0.58
0.56
0.54
0.52
0.5
0.48
0.46
0.1
0.2
0.3
0.4
0.5
0.6
0.7
0.8
0.9
1

Results

Run	A	CWS	P	R
Test corpus:				
sim-lin	55.3	55.9	53.7	75.5
sim-wn	53.6	55.3	53.4	56.5
Development corpus:				
sim-lin	61.0	64.9	57.6	81.8
sim-wn	63.4	67.4	61.6	70.6

Results

Subtask	A	P	R
CD	84.7	74.7	93.3
IE	55.0	95.0	52.8
MT	46.7	63.3	47.5
QA	42.3	53.9	43.8
RC	49.3	88.6	49.6
PP	42.0	80.0	45.5
IR	53.3	75.6	52.3
Overall	55.3	75.5	53.7

Results

- thresholds very corpus-specific -> very difficult to estimate
- deeper text features for tuning system performance
- no comparsion between similarity measures possible

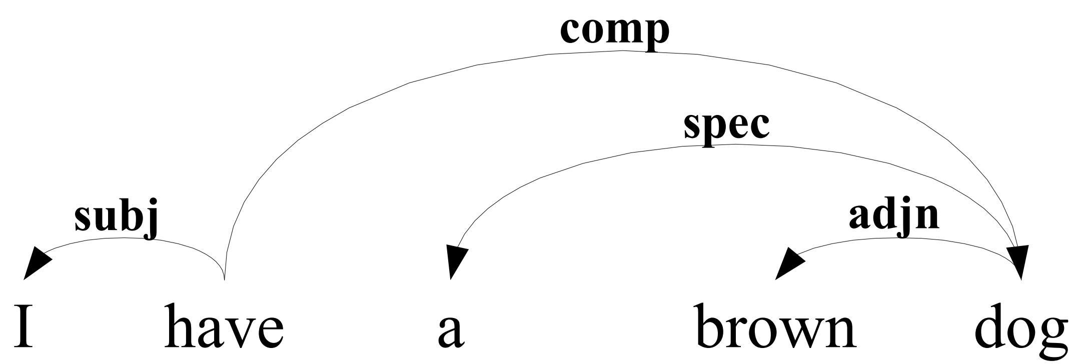
Recognizing Textual Entailment with Tree Edit Distance (Kouylekov & Magnini 2005)

Method:

- tree edit distance algorithm on dependency trees

Dependency Trees (MINI-PAR; Lin 1998b)

1)

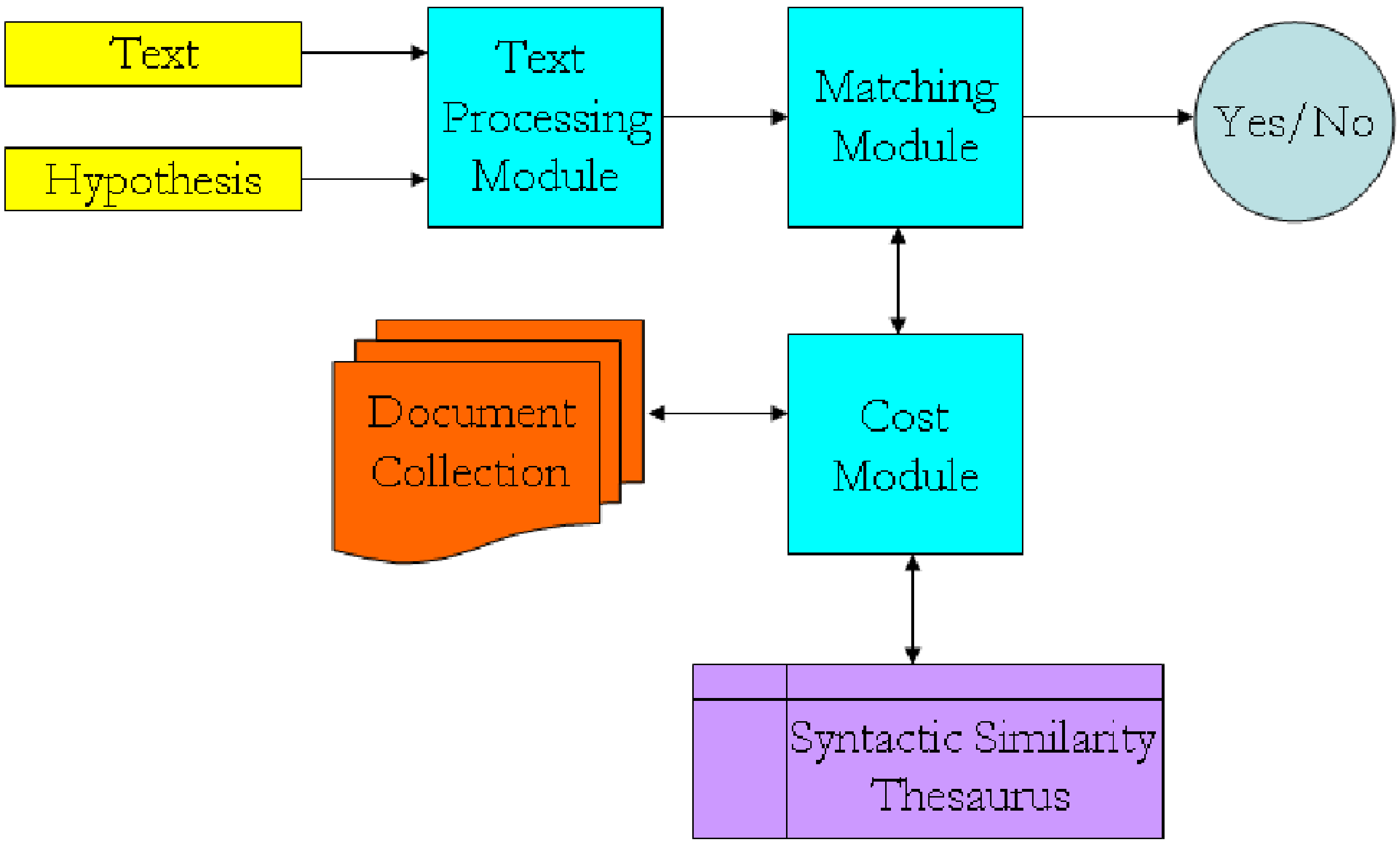


2)

Key

word	cate	head	rel
Kim	N	< promised	subj
promised	V		
Alex	N	> promised	obj1
to	Aux	< bring	aux
bring	V	> promised	obj2
some	Det	< wine	spec
wine	N	> bring	obj1

System Architecture (1/4)



System Architecture (2/4)

Text Processing Module

- sentence splitting
- dependency parsing

Matching Module

- finding best sequence of editing operations

Tree Editing Operations

Matching
Module

Insertion

- insert a node from H to T
- label with source label

Deletion

- delet a node in T
- attach all its children to the parent of the node

Substitution

- change the label of a node in the source tree into a label of a node in the target tree, if the nodes share the same category
- replace the relation of the old node with the relation of the new node

System Architecture (3/4)

Cost Module

return the cost of an edit operation

Cost of

- Insertion : $\text{ins}(w) = \text{idf}(w)$
- Deletion : 0
- Substitution : $\text{subs}(w1,w2) = \text{ins}(w2) * (1 - \text{sim}(w1,w2))$

System Architecture (4/4)

Yes/No

Global entailment scoring

$$score(T,H) = \frac{ed(T,H)}{ed(,H)}$$

$ed(T,H)$: edit distance cost

Recognizing Textual Entailment with Tree Edit Distance

Results

run	measure	CD	IE	MT	QA	RC	PP	IR	Overall
1	accuracy	0.78	0.48	0.50	0.52	0.52	0.52	0.47	0.55
	cws	0.89	0.50	0.55	0.49	0.53	0.48	0.51	0.60
	precision								0.55
	recall								0.64
2	accuracy	0.78	0.53	0.49	0.48	0.54	0.48	0.47	0.55
	cws	0.89	0.53	0.53	0.42	0.58	0.43	0.50	0.58
	precision								0.56
	recall								0.50

run 1 : edit-distance approach for all tasks

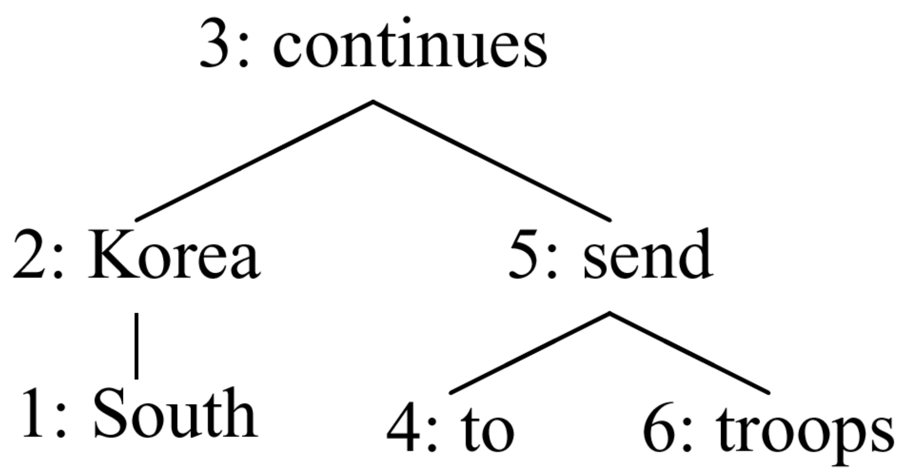
run 2 : edit-distance approach for CD task
for all other tasks linear sequence of words

Results & Future Works

- slightly better results for edit-distance approach
- integration of specialized name entity recognizer
- integration of resources (e.g. entailment patterns, WordNet)
- improvements of the tree-edit distance algorithm
- improvement of the dependency parser

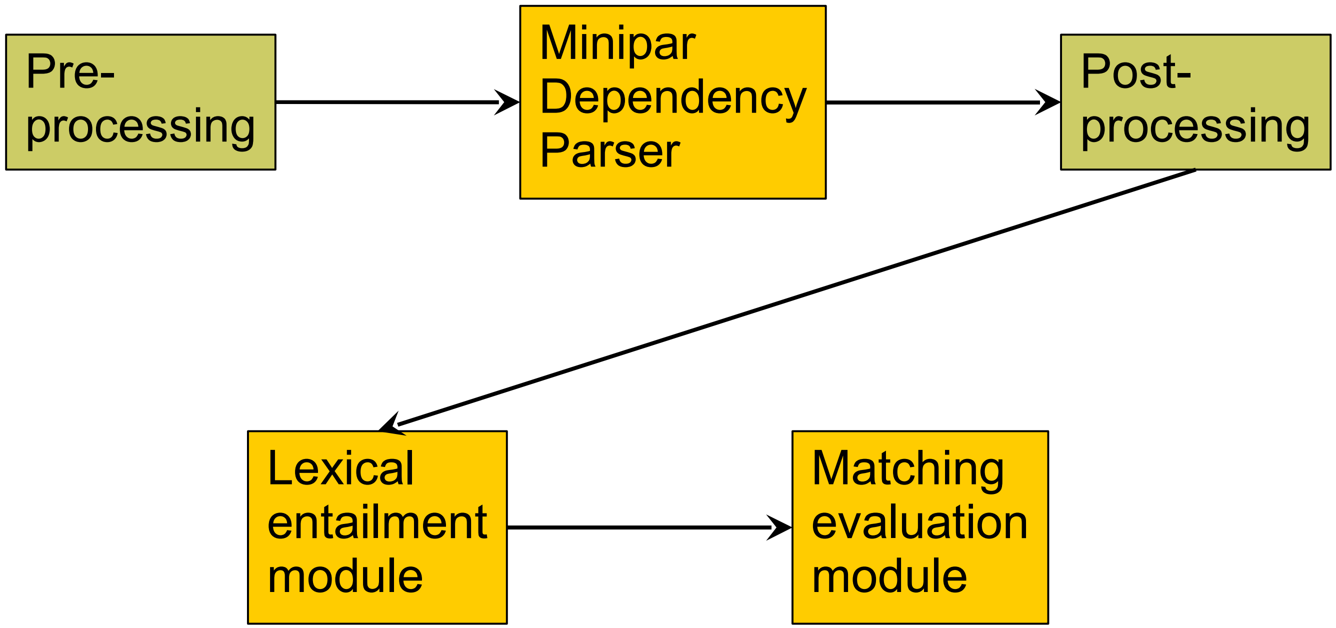
Textual Entailment Recognition Based on Dependency Analysis and WordNet
(Herera et al. 2005)

- Method:
  - dependency tree transformation
  - matching-based



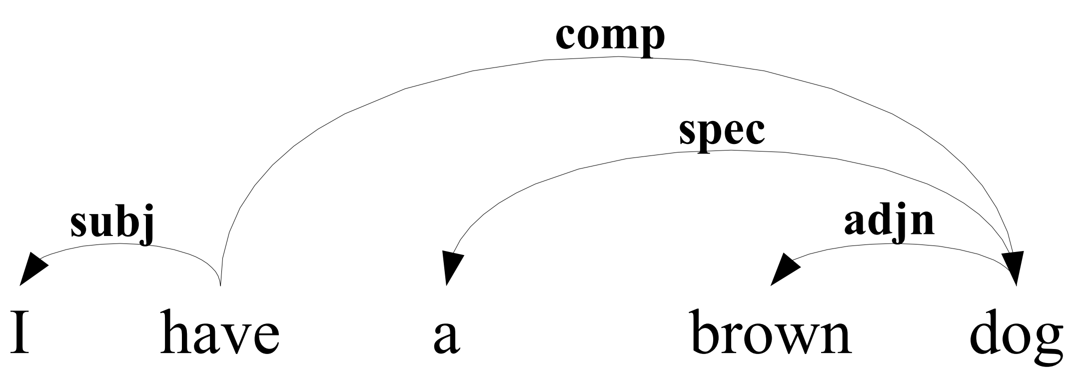
Example: Dependency tree of the sentence:
South Korea continues to send troops.

System Architecture



System Architecture

- MINIPAR Dependency Parser
  - a principle-based
  - broad-coverage parser



System Architecture

- Lexical Entailment Module

  T lexically entails *H*, if *T* is a node in the dependency tree whose lexical unit entails the lexical unit of the node *H* in the hypothesis' dependency tree.

  - WordNet Relations (Synonymy, Similarity, Hyponymy, Entailment, Antonymy)
  - WordNet Multiword Recognition
  - Negation

  Output: pairs(T,H), where T is a node whose lexical unit entails the lexical unit of the node H.

System Architecture

Lexical Entailment Module /Post processing

- WordNet Multiword Recognition

  If two strings differ in less than 10%, the candidate matches the WordNet multiword.

  E.g.: Japanise_capital – Japanese_capital

Levenshtein edit distance:
The Levenshtein distance between two strings is given by the minimum number of operations needed to transform one string into the other, where an operation is an insertion, deletion, or substitution of a single character

Lexical Entailment Module

- WordNet relations
  - Synonymy / Similarity

    lexical unit T entails lexical unit H, if the WordNet synonymy relation holds or a similarity relation holds between them.

    E.g.: discover - reveal
    obtain - receive
    lift – rise
    allow - grant

Lexical Entailment Module

- WordNet relations
  - Hyponymy / WordNet Entailment

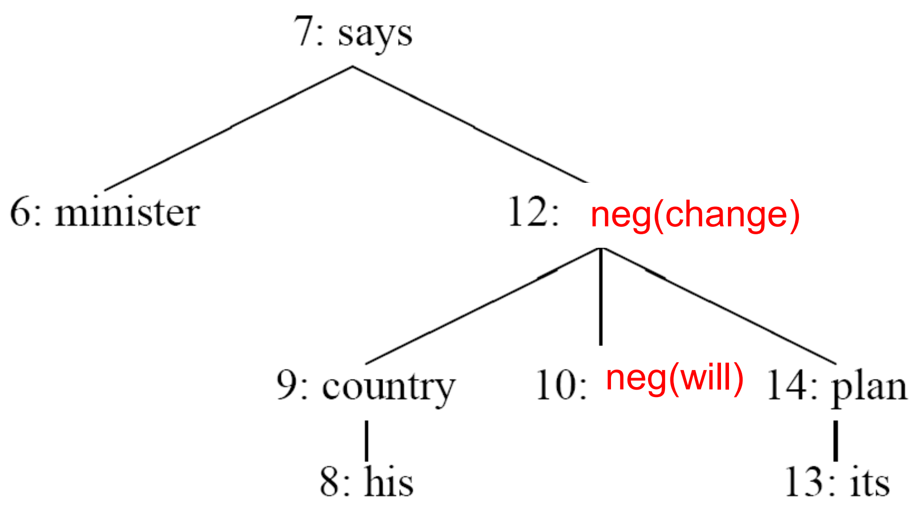
    lexical unit T entails lexical unit H if there is a path from one synset of T to one synset of H with hyponymy and WordNet Entailment relations between intermediate synsets.

    E.g.: glucose – sugar
    crude – oil
    death - kill

System Architecture

Lexical Entailment Module

- Negation / Antonymy
  - Entailment is not possible between a lexical unit and its negation.
  - The negation relation is propagated to its antecestor until the head.

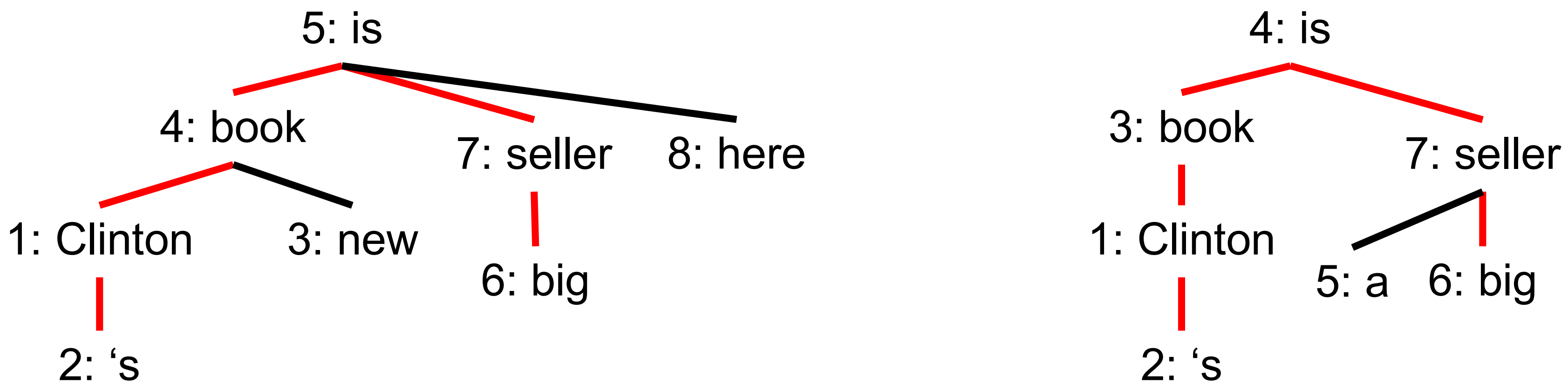


E.g.: neg(change)
-> continue

System Architecture

- Matching Evaluation Module

  Similarity between text and hypothesis is defined as the proportion of hypothesis' nodes pertaining to matching branches in the dependency tree of text.



threshold: 50%
for training corpus
RTE I

Results (1/2)

Development phase

	Accuracy	
Baseline I — word overlap	76.26% (CD)	54.95% (all)
Baseline II — lemma overlap with WordNet	71.13% (CD)	55.48% (all)
Proposed system	80.41% (CD)	56.36% (all)

Accuracy for particular application settings: 46.67% - 55.56%

Results (2/2)

Challenge Results

	Accuracy	
Baseline III lemma overlap without WordNet	79.33% (CD)	55.75% (all)
Modified Proposed System with subj/obj relations	78.67 (CD)	54.75% (all)

Accuracy for particular application settings: 42.55% - 55.38%

Conclusions

- the presented matching-based approach is not appropriate for the textual entailment problem
- some kind of paraphrasing useful
- high matching between nodes in H and T, but H branches match with disperse T's branches

Comparsion / Comments

- all systems use word overlap
- thresholds very corpus dependent
- systems, which use more than word overlap information only slightly better
- the systems do best for the CD task
- use of entailment patterns / paraphrases for system improvement

Shallow Methods in RTE References

- I. Dagan, O. Glickman and B. Magnini: The PASCAL Recognising Textual Entailment Challenge, in RTE-1, 2005.
- Dekang Lin. 1998a. An information-theoretic definition of similarity. In *Proceedings of International Conference on Machine Learning.*
- Dekang Lin. 1998b. *Dependency-based Evaluation of MINIPAR*. Workshop on the Evaluation of Parsing Systems, Granada, Spain, May, 1998.
- J. Herrera, A. Penas, F. Verdejo: Textual Entailment Recognision Based on Dependency Analysis and WordNet, in: RTE-1, 2005.
- G. Hirst and D. St-Onge . 1998 : Lexical chains as representation of context for the detection and correction of malapropisms. In Fellbaum Christiane, editor, *WordNet: An electronic lexical database*. The MIT Press.
- V. Jijkoun and M. de Rijke: Recognizing Textual Entailment Using Lexical Similarity, in: RTE-1, 2005.
- M. Kouylekov and B. Magnini: Recognizing Textual Entailment with Tree Edit Distance Algorithms, in: RTE-1, 2005.
- Ch. Monz and M. de Rijke: Light-Weight Entailment Checking for Computational Semantics, in: ICoS-3 2001.